

BUNDESREPUBLIK DEUTSCHLAND

DE00/77

EU

Klemm



REC'D	28 FEB 2000
WIPO	PCT

091890432

Bescheinigung

Die Siemens Aktiengesellschaft in München/Deutschland hat eine Patentanmeldung unter der Bezeichnung

"Verfahren zum gesicherten Zugriff auf zumindest eine Variable in einem präemptiv Multitasking-gesteuerten Prozessorsystem"

am 29. Januar 1999 beim Deutschen Patent- und Markenamt eingereicht.

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

Die Anmeldung hat im Deutschen Patent- und Markenamt vorläufig das Symbol G 06 F 9/46 der Internationalen Patentklassifikation erhalten.

München, den 16. Februar 2000

Deutsches Patent- und Markenamt

Der Präsident

Im Auftrag

199 03 599.7

Hoiß



PRIORITY DOCUMENT

SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)

THIS PAGE BLANK (USPTO)

Beschreibung

5 Verfahren zum gesicherten Zugriff auf zumindest eine Variable in einem präemptiv Multitasking-gesteuerten Prozessorsystem

In bestehenden und zukünftigen Informationsverarbeitungsanlagen, z.B. Personal Computern, werden mit Hilfe des Betriebssystems,

10 Softwareobjekte - üblicherweise auch als Prozesse bezeichnet - so verwaltet, daß das Hardwaresystem, insbesondere die in der Informationsverarbeitungsanlage vorgesehene prozeßverarbeitende Einrichtung, z.B. der Prozessor, mit dem Ziel hoher Gesamteffizienz gleichmäßig ausgelastet wird. So

15 werden die durch das Betriebssystem dem Prozessor zugeteilten Softwaremodule - üblicherweise auch als Tasks bezeichnet - durch den Prozessor bearbeitet. Hierbei sind für die einen Monoprozessor aufweisenden Informationsverarbeitungsanlagen,

d.h. die Informationsverarbeitungsanlage besitzt nur einen 20 Prozessor, spezielle Betriebssysteme, z.B. Windows 95, vorsehen, die auch auf einem Monoprozessor einen Mehrbenutzer- und/oder einen Mehrprozeßbetrieb ermöglichen - siehe hierzu insbesondere „Architektur von Betriebssystemen“, H. Wetterstein, Hanser Studien Bücher, 1984, S. 54 u. ff.. Der für den

Mehrprozeßbetrieb eines Prozessors erforderliche Betriebsmodus ist in der Fachwelt unter dem Begriff „Mehrprogrammbetrieb“ („multiprogramming“) oder auch „Multitasking“ bekannt. Somit kann durch die Informationsverarbeitungsanlage, während 30 der Ausführung einer Task auch eine weitere Task wie z.B. das Lesen von Daten von einem Speichermedium der Informationsverarbeitungsanlage oder z.B. das Anzeigen von Daten auf einer Datensichtstation „quasiparallel“ ausgeführt werden.

Weiterhin wird zwischen „kooperativen“ und „präemptiven“ Multitasking unterschieden. Beim „kooperativen“ Multitasking bestimmt jede einzelne aktuell ausgeführte Task die Zeitdauer 35 der Beanspruchung des Prozessors nach Bedarf selbst, d.h. die

aktuell ablaufende Task entscheidet über den Zeitpunkt der Freigabe des Prozessors für die Verarbeitung weiterer Tasken. Beim „präemptiven“ Multitasking wird durch eine Task des Betriebssystems, in der Fachwelt als „Scheduler“ oder auch 5 „Taskscheduler“ bekannt, die aktuell ausgeführte Task nach Beendigung einer vordefinierten bzw. zugeteilten Zeitdauer unterbrochen, d.h. der Zeitpunkt der Zuteilung und der Freigabe des Prozessors wird mit Hilfe des Taskschedulers durchgeführt.

10

Für die Ausführung einer Aufgabe des Betriebssystems, d.h. beispielsweise einer Betriebssystemtask wie dem Taskscheduler, ist ein spezieller Betriebsmodus des Prozessors zum Schutz der Daten der Betriebssystemtask vorgesehen, der Supervisor- bzw. Kernel-Modus genannt wird – siehe hierzu insbesondere Andrew S. Tanenbaum „Betriebssysteme – Entwurf und Realisierung“ Teil 1, Prentice-Hall International, 1990, S.31/32. Dazu wird der Prozessor mit Hilfe eines Supervisor-Aufrufes von einem User-Modus in den Supervisor-Modus umgeschaltet und somit wird die Kontrolle des Prozessors dem Betriebssystem bzw. dessen Tasks übertragen. Im Unterschied zum Supervisor-Modus sind im User-Modus nicht alle Instruktionen 15 zulässig, unter anderem ist Tasks im User-Modus die Verwendung von Ein- und Ausgabeinstruktionen sowie von einigen speziellen Instruktionen untersagt. Ebenso ist im User-Modus in der Regel nicht der Zugriff auf alle Daten möglich, d.h. beispielsweise sind die Daten des Betriebssystems für Nicht-Betriebssystemtasks weder lesbar noch veränderbar.

30 Speziell bei nach dem Multitasking-Prinzip wirkenden Informationsverarbeitungsanlagen müssen Variablen oder Blöcke von Variablen, auf die bei der Abarbeitung einer Task zugegriffen wird, gegen konkurrierende Zugriffe, beispielsweise durch weitere Tasks, geschützt werden. Hierdurch wird sichergestellt, daß beispielsweise die beim zweifachen gleichzeitigen 35 Variablenzugriff auftretenden Fehler in der zugreifenden Task zu keinen Blockaden von weiteren Tasks oder der gesamten In-

formationsverarbeitungsanlage führen können. Ein derartiger Schutzmechanismus wird im folgenden mit Hilfe der Formulierung „gesicherter Zugriff“ auf zumindest eine Variable beschrieben, wobei der Begriff Variable sowohl eine in einer Speichereinheit gespeicherte Variable eines Softwaremoduls als auch eine in einem Hardwareregister gespeicherte, hardwarebezogene Einstellinformation bezeichnen kann. Derartige gesicherte Zugriffe erfolgen bei bestimmten Problemstellungen, z.B. in Informationsanlagen, die zur Steuerung von Echtzeit-systemen dienen, jedoch auf administrierbare Daten zurück-greifen müssen, häufig und sind von kurzer zeitlicher Dauer im Vergleich zur mittleren zeitlichen Dauer zwischen zwei aufeinanderfolgenden Taskwechseln. Folglich ist die Wahr-scheinlichkeit für einen Taskwechsel während eines gesicher-ten Zugriffs sehr gering, kann aber keinesfalls ausgeschlos-sen werden.

Die Realisierung eines „gesicherten Zugriffs“ durch eine Task kann mit Hilfe von verschiedenen Schutzmechanismen erfolgen.

Hierzu zählt unter anderem das Setzen einer Taskwechselsperre zur Vermeidung eines konkurrierenden Zugriffs durch eine wei-tre Task auf die Variablen, auf die durch die auf dem Pro-zessor aktuell laufende Task zugegriffen wird. Dazu wird vor dem Zugriff auf die zu lesenden Variablen mit Hilfe eines Su-pervisor-Aufrufes der Prozessor in den Supervisor-Modus umge-schaltet und ein Setzen einer Taskwechselsperre beim Be-triebssystem angefordert, um für die aktuell zugreifende Task einen Exklusivzugriff für den Prozessor und somit auch für die gewünschte Variable zu erreichen. Anschließend wird der Prozessor in den User-Modus zurückgeschaltet und durch die zuvor unterbrochene Task kann nun der gewünschte Zugriff auf die Variable gesichert, d.h. ohne Unterbrechung vorgenommen werden. Nach Beendigung des gesicherten Zugriffs durch die aktuell ablaufende Task muß erneut durch einen Supervisor-Aufruf in den Supervisor-Modus gewechselt und in diesem die Taskwechselsperre durch das Betriebssystem zurückgesetzt wer-den. Zur Weiterbearbeitung der aktuell abzuarbeitenden Task

wird anschließend der Prozessor in den User-Modus zurückgesteuert und die beim Setzen der Taskwechselsperre aktivierte Zeitüberwachung zur Vermeidung einer beliebig langen Blockierung des Prozessors deaktiviert.

5

Eine weitere Methode zur Realisierung eines gesicherten Zugriffs wird bei der Synchronisation von Tasks, d.h. der Abstimmung von mehreren auf den Prozessor abwechselnd zugreifenden Tasks, zur Vermeidung der beim Multitaskingbetrieb 10 auftretenden Konflikte eingesetzt. Hierbei wird für die Synchronisation der einzelnen Tasks häufig die Semaphortechnik verwendet. Gemäß seiner mathematisch-theoretischen Definition ist ein Semaphor eine ganzzahlige, nicht negative Variable verbunden mit einer Warteschlange. Hierbei legt der Anfangswert des Semaphors fest, wieviele Tasks sich gleichzeitig in einem durch ein Semaphor kontrollierten gesicherten Abschnitt befinden dürfen. Die Warteschlange enthält die Tasks, die auf ein Betreten des gesicherten Abschnitts warten. Hierzu wird von der aktuell ablaufenden Task zur Realisierung des gesicherten Zugriffs auf eine Variable mittels eines ununterbrechbaren Lese-Schreibzyklus ein Semaphor überprüft und modifiziert. Ist beispielsweise das Semaphor größer als 0, so wird es dekrementiert und anschließend durch die aktuell ablaufende Task der gesicherte Zugriff auf die gewünschte Variable durchgeführt. Ist das Semaphor bereits gleich 0, so wird die einen gesicherten Zugriff anfordernde Task in den Wartezustand gesteuert und die Semaphorvariable nicht geändert. Am Ende des gesicherten Zugriffs auf die Variable wird überprüft, ob Tasks an diesem Semaphor warten und gegebenenfalls eine der im Wartezustand befindlichen Tasks aktiviert, d.h. der Prozessor zugeteilt. Falls keine Task am Semaphor wartet, wird das Semaphor wieder mittels eines ununterbrechbaren Lese-Schreibzyklus inkrementiert. Diese ununterbrechbaren Lese-Schreibzugriffe auf die Semaphorvariable können entweder ähnlich wie bei der Methode der Taskwechselsperre durch einen Supervisor-Aufruf und der anschließenden Behandlung durch das Betriebssystem oder im User-Modus mit spezieller

Unterstützung durch die Prozessor- und Prozessorbus-Hardware realisiert sein. Auch hier ist für die Dauer des gesicherten Zugriffs eine Zeitüberwachung vorgesehen, deren Aufgabe darin besteht eine überdurchschnittlich lange Blockierung des Prozessors zu vermeiden.

Bei den zuvor beschriebenen Realisierungen eines gesicherten Zugriffs auf Variablen sind bei jedem Zugriff mehrere Betriebsmodiwechsel einschließlich der zugehörigen betriebs-technischen Taskbearbeitung oder eine spezielle Unterstützung durch eine Prozessor- und Prozessorbus-Hardware erforderlich, d.h. gesicherte Zugriffe auf Variablen erhöhen die Belastung des Prozessors oder erfordern zusätzliche und speziell unterstützende Hardware.

Die der Erfindung zugrundeliegende Aufgabe besteht darin, die Realisierung eines gesicherten Zugriffs auf zumindest eine Variable in einem präemptiv Multitasking-gesteuerten Prozessorsystem zu verbessern. Die Aufgabe wird durch die Merkmale des Patentanspruches 1 gelöst.

Der wesentliche Aspekt des erfindungsgemäßen Verfahrens ist darin zu sehen, daß zum gesicherten Zugriff auf zumindest eine Variable in einem präemptiv Multitasking-gesteuerten Prozessorsystem ein Zugriffsstatusspeicher vorgesehen ist, in den vor einem aktuellen Zugriff auf zumindest eine Variable von der zugreifenden Task eine Sperrinformation eingetragen wird. Weiterhin wird bei einem von dem Taskscheduler während des aktuellen Zugriffs beabsichtigter Taskwechsel mit dessen Hilfe in den Zugriffsstatusspeicher eine Taskwechselinformation eingetragen. Am Ende des aktuellen Zugriffs wird eine Freigabeinformation in den Zugriffsstatusspeicher eingetragen und bei eingetragener Taskwechselinformation der angeforderte Taskwechsel durch die aktuell zugreifende Task eingeleitet.

Die Verwendung eines zusätzlichen Zugriffsstatusspeichers bringt den Vorteil mit sich, daß das beispielsweise bei der Methode der Taskwechselperre erforderliche Umschalten des

Prozessors in den Supervisor-Modus und das anschließende Ausführen einer Betriebssystemtask entfällt und somit eine erhebliche dynamische Entlastung des Prozessors erreicht wird, zumal gesicherte Zugriffe auf Variablen bei bestimmten, beim 5 Betrieb einer Informationsverarbeitungsanlage auftretenden Problemstellungen sehr häufig stattfinden. Zusätzlich erfordert das Eintragen der Sperrinformation, der Taskwechselinformation oder der Freigabeinformation nur wenige Maschinenbefehle und ist somit einfach programmtechnisch realisierbar. 10 Des Weiteren ist beim erfindungsgemäßen Verfahren im Gegensatz zur Semaphortechnik keine zusätzliche Hardwareunterstützung in Form von Prozessor- oder Prozessorbus-Hardware nötig, was zu einer kostengünstigen und nicht an spezielle Hardware gebundenen Realisierung des gesicherten Zugriffs auf Variablen 15 führt.

Gemäß einer weiteren Ausgestaltung des erfindungsgemäßen Verfahrens wird vor einem Taskwechsel durch den Taskscheduler der Zugriffsstatusspeicher auf eine eingetragene Sperrinformation überprüft und bei eingetragener Sperrinformation wird durch den Taskscheduler der beabsichtigte Taskwechsel verzögert - Anspruch 2. Hierdurch wird vorteilhaft während des gesicherten Zugriffs die zugreifende Task nicht durch einen von einer weiteren Task beabsichtigten Taskwechsel unterbrochen 25 und zusätzlich wird der beabsichtigte Taskwechsel nicht verworfen, sondern verzögert, so daß nach der Auswertung der Taskwechselinformation am Ende des gesicherten Zugriffs der beabsichtigte Taskwechsel durch den Taskscheduler unmittelbar nachgeholt werden kann.

30 Ein weiterer wesentlicher Aspekt des erfindungsgemäßen Verfahrens ist darin zu sehen, daß zusätzlich zum Eintragen der Taskwechselinformation eine Zeitüberwachung mit einer mindestens die Dauer des gesicherten Zugriffs umfassenden Zeitspanne aktiviert wird und daß nach Ablauf der definierten Zeitspanne der aktuelle Zugriff beendet wird - Anspruch 3. 35 Vorteilhaft wird die Zeitüberwachung beim erfindungsgemäßen

Verfahren nicht generell beim Initialisieren eines gesicherten Zugriffs, sondern nur bei einem während des aktuellen Zugriffs beabsichtigten Taskwechsels aktiviert und damit entfällt die dynamische Belastung, die üblicherweise bei der

5 Verwendung der bereits bekannten Verfahren wie z.B. Semaphorteknik oder das Setzen einer Taskwechselperre erforderlich ist. Dies führt zu einer zusätzlichen dynamischen Entlastung der Informationsverarbeitungsanlage bzw. des Prozessors.

10 Nach einer weiteren Ausgestaltung des erfindungsgemäßen Verfahrens wird am Ende des gesicherten Zugriffs und vor dem Eintragen der Freigabeinformation der Inhalt des Zugriffsstatusspeichers überprüft, so daß beim Vorliegen einer Taskwechselinformation die aktivierte Zeitüberwachung deaktiviert und

15 von der aktuell zugreifenden Task eine betriebstechnische, den beabsichtigten Taskwechsel einleitende Information an den Taskscheduler übermittelt wird - Anspruch 4. Vorteilhaft ist durch die Überprüfung des Inhalts des Zugriffsstatusspeichers sichergestellt, daß unmittelbar nach Beendigung des gesicherten Zugriffs, der durch die Taskwechselinformation angezeigte, beabsichtigte Taskwechsel dem Taskscheduler mitgeteilt wird, da ohne das Anzeigen der betriebstechnischen, den beabsichtigten Taskwechsel anzeigenenden Information der Taskscheduler den verzögerten Taskwechsel nicht ausführen würde.

20 Statt dessen würde der beabsichtigte Taskwechsel zu dem Zeitpunkt ausgeführt, an dem die aktuell zugreifende Task durch den Taskscheduler unterbrochen wird, d.h. der beabsichtigte Taskwechsel würde unnötig über die Zeitspanne des gesicherten Zugriffs hinaus verzögert.

30 Weitere vorteilhafte Ausgestaltungen des erfindungsgemäßen Verfahrens sind den weiteren Ansprüchen zu entnehmen.

35 Im folgenden wird das erfindungsgemäße Verfahren anhand einer Figur näher erläutert.

In Figur 1 sind beispielhaft eine erste und eine zweite Anwendertask T₁, T₂, sowie eine Betriebssystemtask BST gemäß ihrer zeitlichen Bearbeitung durch den Prozessor einer nach dem präemptiven Multitasking Verfahren wirkenden Informationsverarbeitungsanlage dargestellt. Des Weiteren sind ein Supervisor-Modus SM und ein User-Modus UM des Prozessors und die zu gehörigen Tasks durch zwei separate Bereiche angedeutet. Dabei ist im Supervisor-Modus SM die Betriebssystemtask BST, im weiteren auch als Scheduler bzw. Taskscheduler BST bezeichnet, und im User-Modus sind beispielhaft eine erste und eine zweite Anwendertask T₁, T₂ zur Bearbeitung durch den Prozessor dargestellt. Hierbei wird eine aktuell im Wartezustand befindliche Task - beispielsweise in der Figur 1 zum Zeitpunkt null insbesondere die Betriebssystemtask BST und die zweite Anwendertask T₂ - mit Hilfe einer gestrichelten, mit BST und T₂ bezeichneten Linie und eine aktuell ausgeführte Task - in der Figur 1 zum Zeitpunkt null die erste Anwendertask T₁ - durch eine mit T₁ bezeichneten, durchgezogene Linie angedeutet.

20

Zur Darstellung des zeitlichen Ablaufs des erfindungsgemäßen Verfahrens eines gesicherten Zugriffs gz auf zumindest eine Variable ist eine Zeitachse t vorgesehen, worauf ein erster, zweiter, dritter, vierter und fünfter Zeitpunkt t₁, t₂, t_{2'}, t₃, t_{3'} markiert sind. Des Weiteren ist eine Speichereinheit SE1 mit einer Zugriffstatusspeichereinheit ZSE1 zu dem ersten, dritten und vierten Zeitpunkt t₁, t_{2'}, t₃ dargestellt, wobei in der Speichereinheit ZSE1 Informationen bzgl. der ersten, aktuell ablaufenden Task T₁ eingetragen werden, wobei der Speicher beispielsweise als Teilbereich eines flüchtigen Speichers realisiert sein kann. Gemäß dem erfindungsgemäßen Verfahren können in die der ersten, aktuell ablaufenden Anwendertask T₁ zugeteilten Zugriffstatusspeichereinheit ZSE1 unter anderem eine Sperrinformation SI, eine Taskwechselinformation WI und eine Freigabeinformation FI eingetragen werden.

Desweiteren ist die Zeitdauer eines gesicherten Zugriffs gz auf zumindest eine Variable durch die erste Anwendertask T1 dargestellt, die sich vom ersten Zeitpunkt t1 bis zum vierten Zeitpunkt t3 erstreckt. Zum Zeitpunkt null ist die erste Anwendertask T1 bereits aktuell dem Prozessor zugeteilt und die zweite Anwendertask T2, sowie die Betriebssystemtask BST befinden sich im Wartezustand. Zum ersten Zeitpunkt t1 initialisiert die erste Anwendertask T1 einen gesicherten Zugriff auf zumindest eine Variable, d.h. in die Zugriffsstatusspeicherinheit ZSE1(t1) wird durch die erste Anwendertask T1 anstelle der in dieser eingetragenen Freigabeinformation FI die Sperrinformation SI eingetragen. Daraufhin befindet sich die erste, aktuell ausgeführte Anwendertask T1 in einem ununterbrechbaren Ausführungszustand und kann somit gesichert auf die gewünschten Variablen zugreifen.

Zu einem späteren, zweiten Zeitpunkt t2 wird beispielsweise durch ein externes Ereignis EE, z.B. das Vorliegen von externen Nachrichten, oder durch die Überschreitung der vom Taskscheduler BST der ersten Anwendertask T1 zugeteilten Zeitdauer dem Taskscheduler BST eine Taskwechselanforderung TWA angezeigt und daraufhin durch den Taskscheduler BST die aktuell ausgeführte, erste Anwendertask T1 in einen Quasi-Wartezustand wz gesteuert. Anschließend wird durch den Taskscheduler BST noch bevor er nach der erhaltenen Taskwechselanforderung TWA einen Taskwechsel TW anstößt, der Inhalt der Zugriffsstatusspeicherinheit ZSE1(t2') überprüft. Ist in der Zugriffsstatusspeicherinheit ZSE1(t2') für die aktuell ausgeführte, erste Anwendertask T1 eine Sperrinformation SI zu einem dritten Zeitpunkt t2' eingetragen, so wird durch den Taskscheduler BST der angeforderte Taskwechsel TWA verzögert und anstelle der Sperrinformation SI eine Taskwechselinformation WI in die Zugriffstatusspeicherinheit ZSE1(t2') eingetragen. Anschließend wird die erste, aktuell ausgeführte Anwendertask T1 weiterbearbeitet und somit der Quasi-Wartezustand wz durch den Taskscheduler BST wieder beendet. Die erste Anwendertask T1 kann somit den gesicherten Zugriff

(gz) auf die gewünschten Variablen fortführen, ohne das sie vom Taskscheduler BST zur Freigabe des Prozessors gezwungen wird. Zusätzlich wird durch den Taskscheduler BST zum dritten Zeitpunkt t_2' eine Zeitüberwachung TM zur Vermeidung einer 5 unzulässig langen Blockierung des Prozessors durch den gesicherten Zugriff gz der ersten Anwendertask T1 aktiviert.

Am Ende des gesicherten Zugriffs gz - in Figur 1 beispielsweise als vierten Zeitpunkt t_3 angedeutet - wird zunächst der 10 Inhalt der Zugriffstatusspeichereinheit ZSE1(t_3) auf das Vorliegen einer Taskwechselinformation WI überprüft. Ist keine Taskwechselinformation WI in der Zugriffstatusspeichereinheit ZSE1(t_3) eingetragen, so wird durch die aktuell zugreifende, erste Anwendertask T1 anstelle der vorliegenden Sperrinformation SI die Freigabeinformation FI eingetragen und somit der 15 gesicherte Zugriff gz beendet, d.h. die aktuell zugreifende, erste Anwendertask T1 kann daraufhin wieder unterbrochen werden. Die aktuell zugreifende, erste Anwendertask T1 kann im weiteren so lange auf den Prozessor zugreifen bis durch den 20 Taskscheduler BST ein Taskwechsel TW vorgesehen ist, d.h. die der ersten Anwendertask T1 durch den Taskscheduler BST zugeteilte Nutzungszeit des Prozessors abgelaufen ist oder durch ein externes Ereignis EE eine Taskwechselanforderung TWA dem Taskscheduler BST angezeigt wird.

25 Ist dagegen eine Taskwechselinformation WI eingetragen, so wird - wie in Figur 1 dargestellt - dem Taskscheduler BST unmittelbar eine Taskwechselanforderung TWA angezeigt, so daß mit Hilfe dessen nach Abarbeitung der zugehörigen betriebstechnischen Tasks ein Taskwechsel TW durchgeführt werden kann. Zusätzlich wird von der ersten Anwendertask T1 anstelle 30 der eingetragenen Taskwechselinformation WI die Freigabeinformation FI in die Zugriffstatusspeichereinheit ZSE1(t_3) eingetragen und nach Beendigung des gesicherten Zugriffs gz 35 die Zeitüberwachung TM deaktiviert. Weiterhin wird durch den im Supervisor-Modus SM ausgeführten Taskscheduler BST der er-

sten Anwendertask T1 der Prozessor entzogen und diese in den Wartezustand gesteuert.

Anschließend werden im Zeitraum zwischen dem vierten und dem
5 fünften Zeitpunkt t_3, t_3' durch den Taskscheduler BST die für
einen Taskwechsel TW vorgesehenen, betriebstechnischen Tasks
innerhalb des Supervisor-Modus abgearbeitet, d.h. ein
Taskwechsel TW durch das Betriebssystem durchgeführt. Für die
Ausführung der zweiten Anwendertask T2, der der Prozessor ak-
10 tuell zugeteilt wurde, wird der Prozessor in den User-Modus
umgeschaltet und somit kann die zweite Anwendertask T2 ab dem
fünften Zeitpunkt t_3' dem Prozessor zugeteilt werden.

Patentansprüche

1. Verfahren zum gesicherten Zugriff (gz) auf zumindest eine Variable in einem präemptiv Multitasking-gesteuerten Prozessorsystem, wobei für die Bearbeitung der Tasks (T1, T2) ein Taskscheduler (BST) vorgesehen ist,
5 bei dem ein Zugriffsstatusspeicher (ZSE1) vorgesehen ist,
- in den vor einem aktuellen Zugriff (gz) auf zumindest eine Variable von der zugreifenden Task (T1) eine Sperrinformation (SI) eingetragen wird,
10 - in den während des aktuellen Zugriffs (gz) bei einem von dem Taskscheduler (BST) beabsichtigten Taskwechsel (TW) mithilfe einer Taskwechselinformation (WI) eingetragen wird, und
- in den am Ende des aktuellen Zugriffs (gz) eine Freigabeinformation (FI) eingetragen wird und bei eingetragener Taskwechselinformation (WI) der beabsichtigte Taskwechsel (TWA) durch die aktuell zugreifende Task (T1) eingeleitet wird.
15

20 2. Verfahren nach Anspruch 1,
d a d u r c h g e k e n n z e i c h n e t,
daß vor einem Taskwechsel (TW) durch den Taskscheduler (BST)
der Zugriffsstatusspeicher (ZSE1) auf eine eingetragene Sperrinformation (SI) überprüft wird und bei eingetragener Sperrinformation (SI) der Taskscheduler (BST) den beabsichtigten Taskwechsel (TWA) verzögert.

30 3. Verfahren nach Anspruch 1 oder 2,
d a d u r c h g e k e n n z e i c h n e t,
daß zusätzlich zum Eintragen der Taskwechselinformation (WI)
eine Zeitüberwachung (TM) mit einer mindestens die Dauer des gesicherten Zugriffs (gz) umfassenden Zeitspanne aktiviert wird, und daß nach Ablauf der definierten Zeitspanne der aktuelle Zugriff (gz) beendet wird.

35 4. Verfahren nach einem der Ansprüche 1 bis 3,

d a d u r c h g e k e n n z e i c h n e t,
daß am Ende des gesicherten Zugriffs (gz) und vor dem Eintra-
gen der Freigabeinformation (FI) der Inhalt des Zugriffssta-
tusspeichers (ZSE1) überprüft wird, so daß beim Vorliegen ei-
5 ner Taskwechselinformation (WI) die aktivierte Zeitüberwa-
chung (TM) deaktiviert und von der aktuell zugreifenden Task
(T1) eine betriebstechnische, den beabsichtigten Taskwechsel
einleitende Information an den Taskscheduler (BST) übermit-
telt wird.

10

5. Verfahren nach einem der Ansprüche 1 bis 4,

d a d u r c h g e k e n n z e i c h n e t,

daß durch das Eintragen einer Information (SI, WI, FI) in den
Zugriffsstatusspeicher (ZSE1) der Inhalt des Zugriffsstatus-
15 speichers (ZSE1) überschrieben wird.

6. Verfahren nach einem der Ansprüche 1 bis 5,

d a d u r c h g e k e n n z e i c h n e t,

daß die Sperrinformation (SI), die Taskwechselinformation
20 (WI) und die Freigabeinformation (FI) durch zumindest eine
Einbit-Information gebildet werden und die Sperrinformation
(SI) und die Freigabeinformation (FI) in einer gemeinsamen,
einzelnen Zugriffsstatusspeichereinheit (ZSE1) gespeichert
werden.

7. Verfahren nach einem der Ansprüche 1 bis 6,

d a d u r c h g e k e n n z e i c h n e t,

daß eine Variable entweder durch eine in einer Speicherein-
heit gespeicherte Variable eines Softwaremoduls oder durch
30 eine in einem Hardwareregister gespeicherte, hardwarebezogene
Einstellinformation repräsentiert ist.

Zusammenfassung

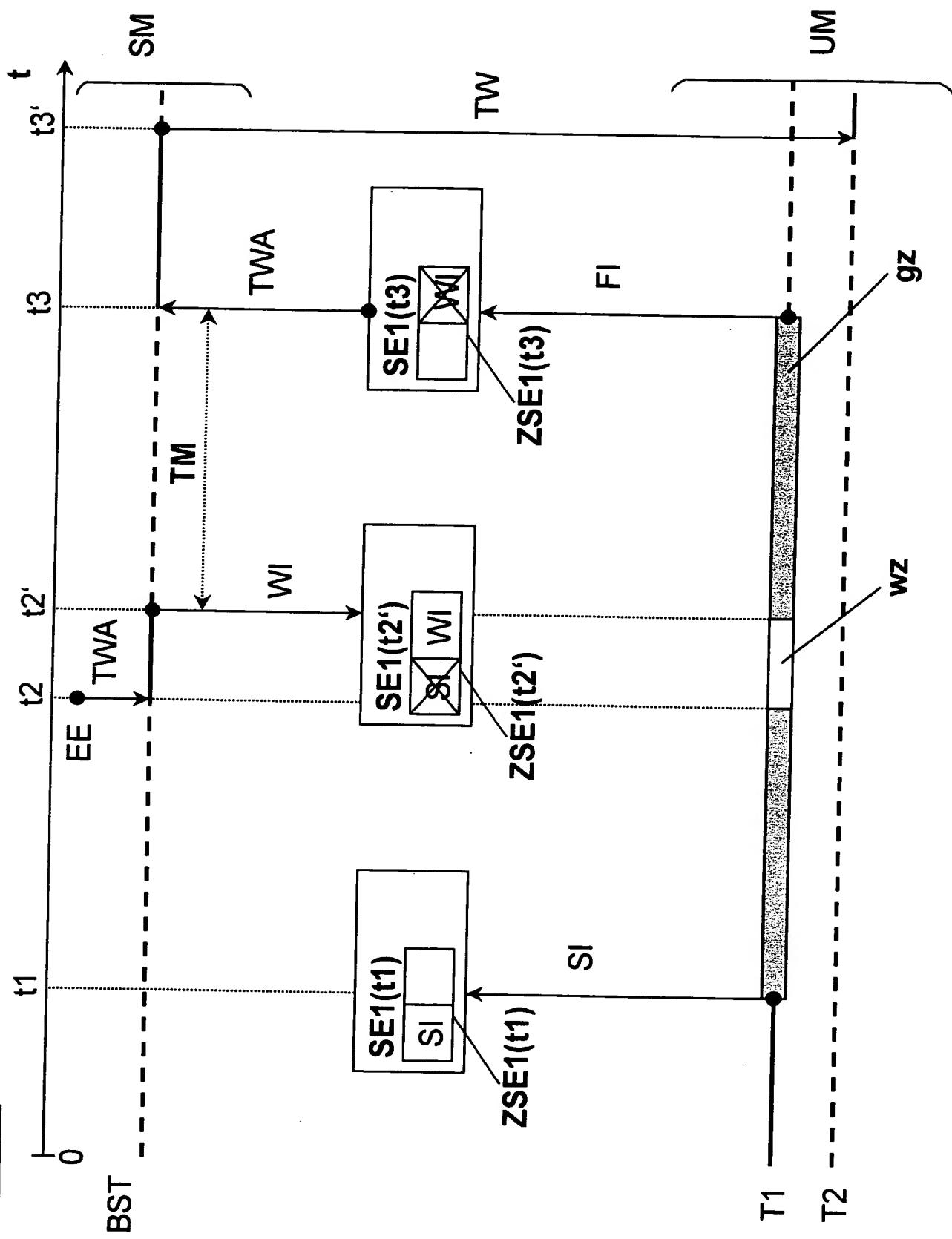
Verfahren zum gesicherten Zugriff auf zumindest eine Variable in einem präemptiv Multitasking-gesteuerten Prozessorsystem

5

In einen Zugriffsstatusspeicher (ZSE1) wird vor einem aktuellen Zugriff auf zumindest eine Variable von der zugreifenden Task (T1) eine Sperrinformation (SI) eingetragen. Weiterhin 10 wird bei einem von einem Taskscheduler (BST) während des gesicherten, aktuellen Zugriffs beabsichtigten Taskwechsels (TWA) mit dessen Hilfe in den Zugriffsstatusspeicher (ZSE1) eine Taskwechselinformation (WI) eingetragen. Am Ende des aktuellen Zugriffs wird eine Freigabeinformation (FI) in den 15 Zugriffsstatusspeicher (ZSE1) eingetragen und bei eingetragener Taskwechselinformation (WI) der verzögerte Taskwechsel (TWA) durch die aktuell zugreifende Task (T1) eingeleitet.

20 Figur

FIG 1



THIS PAGE BLANK (USPTO)

THIS PAGE BLANK (USPTO)